

**CSCI Integration Test (CIT) Procedures**

**Data Bank Shuttle Automated Function Executive (DBSAFE)**

**Checkout and Launch Control Systems (CLCS)**

**84K06581-000-02**

Approval:

\_\_\_\_\_  
Joe Regan, Test Conductor      Date  
DBSAFE CSCI

\_\_\_\_\_  
System Test      Date

\_\_\_\_\_  
Jerry D. Murr, Lead,      Date  
DBSAFE CSCI

\_\_\_\_\_  
Quality Assurance      Date

\_\_\_\_\_  
Larry Wilhelm, Chief      Date  
Software Design Division

\_\_\_\_\_  
Date

NOTE: See "Supporting Document Note" on following page

**PREPARED BY:**

---

---

---

---

---

---

---

---

**Supporting Document Note:** Acronyms and definitions of many common CLCS terms may be found in the following documents: CLCS Acronyms 84K00240 and CLCS Project Glossary 84K00250.

**REVISION HISTORY**

<b>REV</b>	<b>DESCRIPTION</b>	<b>DATE</b>
Basic	CLCS Redstone Delivery CSCI Integration Test Procedures for DBSAFE - Document, 84K06515	10/21/97
A	CLCS Thor Delivery CSCI Integration Test Procedures for DBSAFE - Document, 84K06581-000-02	01/27/98

LIST OF EFFECTIVE PAGES				
Dates of issue of change pages are:				
Page No.	A or D*	Issue or Change No.	CR No.	Effective Date**

## Table of Contents

<b>1. SCOPE .....</b>	<b>1-1</b>
1.1 IDENTIFICATION .....	1-1
1.2 PURPOSE .....	1-1
1.3 CSCI OVERVIEW .....	1-1
1.4 HARDWARE AND SOFTWARE CONFIGURATIONS .....	1-2
1.5 DOCUMENT ORGANIZATION .....	1-2
<b>2. APPLICABLE DOCUMENTATION.....</b>	<b>2-1</b>
2.1 PARENT DOCUMENTS .....	2-1
2.2 APPLICABLE DOCUMENTS .....	2-1
2.3 REFERENCE DOCUMENTS .....	2-2
<b>3. TEST CASE DESCRIPTION.....</b>	<b>3-3</b>
3.1 TEST CASE 3.1 - TEST SSA1/SSA2 DATA TYPES .....	3-3
3.1.1 <i>Test Description</i> .....	3-3
3.1.1.1 Detailed Description .....	3-3
3.1.1.2 Resource Requirements .....	3-3
3.1.1.2.1 Test Personnel .....	3-3
3.1.1.2.2 Hardware .....	3-3
3.1.1.2.3 Software .....	3-3
3.1.1.2.4 Data .....	3-3
3.1.1.3 Requirements Summary.....	3-4
3.1.2 <i>Pass/Fail Criteria</i> .....	3-5
3.1.3 <i>Procedure</i> .....	3-5
3.2 TEST CASE 3.2 - TEST SUMMARY CONSTRAINT FDS .....	3-10
3.2.1 <i>Test Description</i> .....	3-10
3.2.1.1 Detailed Description .....	3-10
3.2.1.1.1 Test Personnel .....	3-10
3.2.1.1.2 Hardware .....	3-10
3.2.1.1.3 Software .....	3-10
3.2.1.1.4 Data .....	3-10
3.2.1.2 Requirements Summary.....	3-10
3.2.2 <i>Pass/Fail Criteria</i> .....	3-11
3.2.3 <i>Procedure</i> .....	3-11
3.3 TEST CASE 3.3 - TEST ENUMERATED DATA TYPES .....	3-14
3.3.1 <i>Test Description</i> .....	3-14
3.3.1.1 Detailed Description .....	3-14
3.3.1.2 Resource Requirements .....	3-14
3.3.1.2.1 Test Personnel .....	3-14
3.3.1.2.2 Hardware .....	3-14
3.3.1.2.3 Software .....	3-14
3.3.1.2.4 Data .....	3-14
3.3.1.3 Requirements Summary.....	3-15
3.3.2 <i>Pass/Fail Criteria</i> .....	3-15
3.3.3 <i>Procedure</i> .....	3-15
3.4 TEST CASE 3.4 - TEST COMBINED OI, GPC, PLDOI.....	3-22
3.4.1 <i>Test Description</i> .....	3-22
3.4.1.1 Detailed Description .....	3-22
3.4.1.2 Resource Requirements .....	3-22
3.4.1.2.1 Test Personnel .....	3-22
3.4.1.2.2 Hardware .....	3-22
3.4.1.2.3 Software .....	3-22
3.4.1.2.4 Data .....	3-22

3.4.1.3 Requirements Summary.....	3-22
3.4.2 <i>Pass/Fail Criteria</i> .....	3-23
3.4.3 <i>Procedure</i> .....	3-23
3.5 TEST CASE 3.5 - TEST DUAL ADDRESSES .....	3-26
3.5.1 <i>Test Description</i> .....	3-26
3.5.1.1 Detailed Description .....	3-26
3.5.1.2 Resource Requirements .....	3-26
3.5.1.2.1 Test Personnel .....	3-26
3.5.1.2.2 Hardware .....	3-26
3.5.1.2.3 Software .....	3-26
3.5.1.2.4 Data .....	3-26
3.5.1.3 Requirements Summary.....	3-26
3.5.2 <i>Pass/Fail Criteria</i> .....	3-27
3.5.3 <i>Procedure</i> .....	3-27
3.6 TEST CASE 3.6 - TEST CONVERTED TYPE FIELD .....	3-30
3.6.1 <i>Test Description</i> .....	3-30
3.6.1.1 Detailed Description .....	3-30
3.6.1.2 Resource Requirements .....	3-30
3.6.1.2.1 Test Personnel .....	3-30
3.6.1.2.2 Hardware .....	3-30
3.6.1.2.3 Software .....	3-30
3.6.1.2.4 Data .....	3-30
3.6.1.3 Requirements Summary.....	3-30
3.6.2 <i>Pass/Fail Criteria</i> .....	3-31
3.6.3 <i>Procedure</i> .....	3-31
3.7 TEST CASE 3.7 - TEST CALIBRATION FDS .....	3-34
3.7.1 <i>Test Description</i> .....	3-34
3.7.1.1 Detailed Description .....	3-34
3.7.1.1.1 Test Personnel .....	3-34
3.7.1.1.2 Hardware .....	3-34
3.7.1.1.3 Software .....	3-34
3.7.1.1.4 Data .....	3-34
3.7.1.2 Requirements Summary.....	3-34
3.7.2 <i>Pass/Fail Criteria</i> .....	3-35
3.7.3 <i>Procedure</i> .....	3-35
3.8 TEST CASE 3.8 - TEST SYSTEM FDS .....	3-38
3.8.1 <i>Test Description</i> .....	3-38
3.8.1.1 Detailed Description .....	3-38
3.8.1.2 Resource Requirements .....	3-38
3.8.1.2.1 Test Personnel .....	3-38
3.8.1.2.2 Hardware .....	3-38
3.8.1.2.3 Software .....	3-38
3.8.1.2.4 Data .....	3-38
3.8.1.3 Requirements Summary.....	3-38
3.8.2 <i>Pass/Fail Criteria</i> .....	3-39
3.8.3 <i>Procedure</i> .....	3-39
3.9 TEST CASE 3.9 - TEST IVHM FDS .....	3-42
3.9.1 <i>Test Description</i> .....	3-42
3.9.1.1 Detailed Description .....	3-42
3.9.1.2 Resource Requirements .....	3-42
3.9.1.2.1 Test Personnel .....	3-42
3.9.1.2.2 Hardware .....	3-42
3.9.1.2.3 Software .....	3-42
3.9.1.2.4 Data .....	3-42
3.9.1.3 Requirements Summary.....	3-42
3.9.2 <i>Pass/Fail Criteria</i> .....	3-43
3.9.3 <i>Procedure</i> .....	3-43
Appendix A Acronyms and Definitions.....	A-1

Appendix B Requirements Traceability and Test Methods Matrix.....B-1  
Appendix C Resource Requirements .....C-1  
Appendix D Standard Test Operating Procedures .....D-1

## **CSCI INTEGRATION TEST (CIT) PROCEDURES**

### **DATA BANK SHUTTLE AUTOMATED FUNCTION EXECUTIVE (DBSAFE)**

### **CHECKOUT AND LAUNCH CONTROL SYSTEMS (CLCS)**

#### **1. SCOPE**

This document defines the test approach and procedures to be executed for the Thor delivery of the Data Bank Shuttle Automated Function Executive (DBSAFE) by CLCS Software Development. Testing will occur at the Kennedy Space Center in the Processing Control Center (PCC) Shuttle Data Center (SDC).

##### **1.1 IDENTIFICATION**

This document is the Checkout and Launch Control System (CLCS) Thor Delivery CSCI Integration Test Procedures for DBSAFE Document, 84K06581-000-02, Rev A.

##### **1.2 PURPOSE**

The purpose of this document is to define a suite of test procedures that will accurately assess the delivered software to ensure it is functional and meets project commitments for the Thor delivery. The CLCS DBSAFE software is software ported from the SDC DBSAFE software; the ported CLCS DBSAFE software is considered the CLCS DBSAFE Baseline. These test procedures will assess any deltas made to the Redstone baseline of the ported software.

##### **1.3 CSCI OVERVIEW**

CLCS DBSAFE is a comprehensive Checkout and Launch Control System (CLCS) software capability that provides an interactive user interface supporting the evaluation, incorporation, and historical tracking of engineering changes to the FD Database. DBSAFE for CLCS is ported code baselined from the DBSAFE software developed for the replatform of CCMS Support Software to the Shuttle Data Center (SDC).

The core purpose of the CLCS DBSAFE is to provide the capability to maintain the FD Database. The FD Database is the portion of the CLCS DBSAFE database that contains the information on the measurements, commands, and system parameters needed for CLCS. The attributes of measurements and commands for the orbiters, payloads, ground support equipment, etc., are collected from the various design agencies, processed into a format that is compatible with CLCS, and stored in the FD Database using CLCS DBSAFE software. The data is then available to support the CLCS Application Software Development Environment and Test Build processes.



CLCS DBSAFE also provides the capability to create and maintain TCID build specifications for the FD Directory Build process. CLCS DBSAFE validates and stores user specifications in the CLCS DBSAFE database. CLCS DBSAFE facilitates the generation of TCID specifications by automating the following functions:

- Assignment and traceability of Vehicle Configuration Names (VCN) and formats to Test Configuration Identifiers (TCID) based on a list of engineering provided by Ground Software Integration (GSI)
- Assignment of projected VCNs and formats to TCIDs based on matching each mission/TCID configuration to the effectivities of engineering changes in the FD Database
- Assignment and traceability of format revisions to each mission/TCID based on the format engineering defined in the Shuttle Data Tape (SDT)
- Support elimination of invalid/duplicate/overlapping addressing in the FD Database, that would otherwise cause errors in a TCID build.

## **1.4 HARDWARE AND SOFTWARE CONFIGURATIONS**

CLCS DBSAFE software executes primarily on the LPS Software Development Network (LSDN) of Hewlett-Packard UNIX workstations. The CLCS DBSAFE menus and all user interface applications are written using Oracle\*Forms. All reports are initiated from Oracle\*Forms based applications accessed from the CLCS DBSAFE menu. The reports are written in Oracle\*Report and are generated as background processes running on the LSDN workstation.

The CLCS DBSAFE main menu is initiated through the Relational Software Interface (RSI) system resident on the LSDN. RSI is one of several configuration management tools and techniques designed to fulfill the access control and data integrity requirements of CLCS applications. RSI also satisfies the Ad Hoc Query requirements for CLCS DBSAFE by providing read-only access to the CLCS DBSAFE database via either Oracle SQL\*Plus or Oracle Browser.

All CLCS DBSAFE data is stored in an Oracle database located on a Digital UNIX server within the SDC. For testing purposes, it is assumed that all prerequisite CLCS DBSAFE control and legal values data has been loaded into the database. Data supporting the CLCS DBSAFE management services is located in the same physical database as the FD Database; the term "CLCS DBSAFE database" encompasses all. The distinction is important because CLCS DBSAFE supports a much higher degree of configuration management and control over the data within the FD Database. CLCS DBSAFE uses stored database procedures and triggers to satisfy many processing requirements. Communication between the software running on the LSDN and the database in the SDC is handled by Oracle SQL\*Net.

## **1.5 DOCUMENT ORGANIZATION**

This document is divided into three sections and four appendices:

Section 1, Scope, discusses the purpose of the CSCI Integration Test, provides a system overview , and describes software and hardware configurations for the system.

Section 2, Applicable Documents, lists the documents used to create and those supporting this document.

Section 3, Test Case Description, contains a description of the test cases, the pass/fail criteria, and the procedures in detail.

Appendix A, Acronyms and Definitions, contains a listing of acronyms and selected word definitions (for words which may have multiple interpretations)

Appendix B, Requirements Traceability and Test Methods Matrix, contains the requirements verification matrix for the test.

Appendix C, Resource Requirements, contains a list of software, hardware, and personnel requirements necessary for each test.

Appendix D, Standard Operating Test Procedures, contains any specific, standard procedures identified within the test cases.

## 2. APPLICABLE DOCUMENTATION

The following documents, of the revision shown, form a part of this document to the extent specified.

### 2.1 PARENT DOCUMENTS

The documents in this paragraph establish the criteria and technical basis for the existence of this document. The parent documents are:

Parent Document	Document Number	Rev.	Date
CLCS DBSAFE Software Requirements and Design	84K0910-010	A	1/26/1998

Table 2.1: Parent Documents

### 2.2 APPLICABLE DOCUMENTS

Applicable documents are those documents which form a part of this document. These documents, at the revisions listed below, carry the same weight as if they were stated within the body of this document.

Applicable Document	Document Number	Rev.	Date
CLCS DBSAFE Software Requirements and Design	84K00910-010	A	1/26/1998

Table 2.2: Applicable Documents

## 2.3 REFERENCE DOCUMENTS

Reference documents are those documents which, though not a part of this document, serve to clarify the intent and contents of this document.

Reference Document	Document Number	Rev.	Date
CLCS System Level Specification	84K00200	Basic	6/26/1997
CLCS DBSAFE Software Requirements and Design	84K00910-010	A	1/26/1998

Table 2.3: Reference Documents

### **3. TEST CASE DESCRIPTION**

The test cases will validate that all the stated Thor requirements have been satisfied. This section describes each test case, the expected results, the pass/fail criteria, and a step by step procedure to execute the test. Appendix B contains the Requirements Traceability and Test Methods Matrix, which maps functional requirements to the test case that verifies those requirements. Test cases are stand alone, and can be executed in any order.

#### **3.1 TEST CASE 3.1 - TEST SSA1/SSA2 DATA TYPES**

Validate that the modified support for SSA1/SSA2 data types has been implemented.

##### **3.1.1 Test Description**

###### **3.1.1.1 Detailed Description**

This test case will demonstrate that DBSAFE support for SSA1/SSA2 data types has been modified per the functional requirement stated in section 3.1.1.3. Each step of the test case will demonstrate specific aspects of this support ranging from viewing table structures/data, to manipulating System Status FDs in the database using the DBSAFE forms, to running DBSAFE reports.

###### **3.1.1.2 Resource Requirements**

###### **3.1.1.2.1 Test Personnel**

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

###### **3.1.1.2.2 Hardware**

The following Hardware is required:

- Refer to Section 1.4

###### **3.1.1.2.3 Software**

The following Software is required:

- Refer to Section 1.4

###### **3.1.1.2.4 Data**

The following Data is required:

- Refer to Section 1.4

### 3.1.1.3 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.14	<p>Support for SSA1/SSA2 data types will be modified.</p> <ul style="list-style-type: none"><li>• Remove SCT, HIM, and PCM as valid subtypes for SSA1 type FDs.</li><li>• Add BD as a valid subtype for SSA1 type FDs.</li><li>• Add GMT and HEX as a valid subtype for SSA2 type FDs</li><li>• A SSA2 subtype DEC FD will have a length of 2-16 bits.</li><li>• A SSA2 subtype HEX or GMT FD will have a length of 32 bits.</li><li>• SOURCE will be a required field for both SSA1 and SSA2 type FDs.</li><li>• Remove Bit_Addr, SCT_Offset, SDT_Offset, and Gateway columns from SSA1 and SSA2 type FDs.</li><li>• Add a Subsystem column for both SSA1 and SSA2 type FDs.</li><li>• The valid Subsystems will be GSE, LDB, PCM, EIU, CDL, UPLK, CSG, DDP, CCP, and CCWS</li><li>• Add a new legal table for Subsystems.</li></ul>

### **3.1.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.1.3 Procedure**

Refer to Procedure 3.1 in Table 3.1 for test procedures.

Procedure 3.1 - Test SSA1/SSA2 Data Types					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Via SQL*Plus, a describe will be done on the DBSAFE.SYSTEM_STATUS_HD_INFO table.	The BIT_ADDR, SCT_OFFSET, SDT_OFFSET, and GATEWAY columns will not exist. The SUBSYSTEM column will now exist.			
2.	Via SQL*Plus, a select of all subsystems in DBSAFE.SUBSYSTEM_INFO will be performed.	All the valid subsystems will be present.			
3.	From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and add a SSA1/BD FD to the database. Query the FD back using the Query capability of the Compiler Definition form.	A COMMON_CD_INFO record will exist for the SSA1/BD FD			
4.	From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and add a SSA2/DEC FD to the database. Also demonstrate that a HEX subtype is in the LOV of valid subtypes. Query the FD back using the Query capability of the Compiler Definition form.	A COMMON_CD_INFO record will exist for the SSA2/DEC FD.			



5.	From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and add a SSA2/GMT FD to the database. Query the FD back using the Query capability of the Compiler Definition form.	A COMMON_CD_INFO record will exist for the SSA2/GMT FD.			
6.	From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and add Hardware information for the SSA1/BD FD. Query the FD back using the Query capability of the Hardware Definition form.	A COMMON_HD_INFO record and an associated HD variant record will exist for the SSA1/BD FD.			
7.	From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and add Hardware information for the SSA2/DEC FD. Query the FD back using the Query capability of the Hardware Definition form.	A COMMON_HD_INFO record and an associated HD variant record will exist for the SSA2/DEC FD.			
8.	From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and add Hardware information for the SSA2/GMT FD. Query the FD back using the Query capability of the Hardware Definition form.	A COMMON_HD_INFO record and an associated HD variant record will exist for the SSA2/GMT FD.			

<b>9.</b>	From the Reports option of the CLCS DBSAFE Main Menu, select the Implement suboption, then select the ONEFD suboption. Request a report of the FDs utilized in the above steps.	The report will contain the specified FDs. The SCT/SDT/Bit Address/ Gateway information will not be present in the report. The Subsystem information will be present in the report.			
<b>10.</b>	From the Implement option of the CLCS DBSAFE Main Menu, select the Lock/Review (DB) suboption. Lock the TPS Run ID used to perform the previous steps. This will automatically generate a From-To-Report.	The From-To Report generated by the locking process will not contain the SCT/SDT/Bit Address/ Gateway information. It will contain Subsystem information for the FDs used in the above steps.			

Table 3.1

**End Time:** \_\_\_\_\_

**Signature Page: Test Case 3.1 - Test SSA1/SSA2 Data Types**

<hr/>	<hr/>
<b>Quality Assurance</b>	<b>Date</b>

<hr/>	<hr/>
<b>Test Conductor</b>	<b>Date</b>

**Comments:**

---

---

---

---

---

---

---

---

---

---

## 3.2 TEST CASE 3.2 - TEST SUMMARY CONSTRAINT FDS

Validate that support for Summary Constraint Function Designators has been provided.

### 3.2.1 Test Description

#### 3.2.1.1 Detailed Description

This test case will demonstrate that DBSAFE support for Summary Constraint Function Designators has been provided per the functional requirement stated in section 3.2.1.2. Each step of the test case will demonstrate specific aspects of this support ranging from viewing table structures/data, to manipulating Summary Constraint FDs in the database using the DBSAFE forms.

##### 3.2.1.1.1 Test Personnel

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

##### 3.2.1.1.2 Hardware

The following Hardware is required:

- Refer to Section 1.4

##### 3.2.1.1.3 Software

The following Software is required:

- Refer to Section 1.4

##### 3.2.1.1.4 Data

The following Data is required:

- Refer to Section 1.4

#### 3.2.1.2 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.15	Support for Summary Constraint Function Designators will be provided. <ul style="list-style-type: none"> <li>• Treat Summary Constraint as a new data source in CLCS DBSAFE.</li> <li>• Support all current Pseudo FDs for Summary Constraint data.</li> <li>• Only support compiler and hardware data for Summary Constraint data. (no addressing)</li> </ul>

### **3.2.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.2.3 Procedure**

Refer to Procedure 3.2 in Table 3.2 for test procedures.

Procedure 3.2 - Test Summary Constraint FDs					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Via SQL*Plus , a select of all sources in DBSAFE.SOURCE_INFO will be performed.	A new summary constraint source, SCFD, will be present.			
2.	From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and add a Pseudo FD with a source of SCFD to the database. Query the FD back using the Query capability of the Compiler Definition form.	A COMMON_CD_INFO record will exist for the Pseudo FD with a source of SCFD.			
3.	From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and add Hardware information for the Pseudo FD. Query the FD back using the Query capability of the Hardware Definition form.	A COMMON_HD_INFO record and an associated HD variant record will exist for the Pseudo FD.			
4.	From the Implement option of the CLCS DBSAFE Main Menu, select the Address Definition suboption and attempt to add Address information for the Pseudo FD.	The address information will not be permitted to be entered.			

Table 3.2

End Time: \_\_\_\_\_

**Signature Page: Test Case 3.2 - Test Summary Constraint FDs**

\_\_\_\_\_  
**Quality Assurance**

\_\_\_\_\_  
**Date**

\_\_\_\_\_  
**Test Conductor**

\_\_\_\_\_  
**Date**

**Comments:**

---

---

---

---

---

---

---

---

---

---

### **3.3 TEST CASE 3.3 - TEST ENUMERATED DATA TYPES**

Validate that support for Enumerated data types has been provided.

#### **3.3.1 Test Description**

##### **3.3.1.1 Detailed Description**

This test case will demonstrate that DBSAFE support for Enumerated data types has been provided per the functional requirement stated in section 3.3.1.3. Each step of the test case will demonstrate specific aspects of this support ranging from manipulating Enumerated Class data in the database using the DBSAFE forms, to referencing Enumerated Classes by FDs in the database using the DBSAFE forms, to running DBSAFE reports.

##### **3.3.1.2 Resource Requirements**

###### **3.3.1.2.1 Test Personnel**

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

###### **3.3.1.2.2 Hardware**

The following Hardware is required:

- Refer to Section 1.4

###### **3.3.1.2.3 Software**

The following Software is required:

- Refer to Section 1.4

###### **3.3.1.2.4 Data**

The following Data is required:

- Refer to Section 1.4



### 3.3.1.3 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.16	<p>Support for Enumerated data types will be provided.</p> <ul style="list-style-type: none"><li>• There will be a new form created to allow for definition and maintenance of enumerated class data.</li><li>• There will be DB Revision Tracking associated with enumerated class data.</li><li>• The existing digital pattern types will have the capability to have an associated enumerated class.</li><li>• The existing pseudo digital pattern (PDP) type will have the capability to have an associated enumerated class.</li></ul>

### 3.3.2 Pass/Fail Criteria

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### 3.3.3 Procedure

Refer to Procedure 3.3 in Table 3.3 for test procedures.

Procedure 3.3 - Test Enumerated Data Types					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Adding Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and add an Enumerated Class to the database. As part of the add of the Enumerated Class the following will be verified: <ul style="list-style-type: none"><li>• Verify that the maximum number of elements cannot exceed 2<sup>valen</sup>.</li><li>• Verify that the element values must be between 0 and (2<sup>valen</sup>)-1.</li><li>• Verify that a non-unique element cannot be entered.</li></ul>	An ENUMERATED_CLASS_INFO record and the associated ENUMERATED_ELEMENT_INFO records will exist for the Enumerated Class.			
2.	Adding/Modifying Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and add a second Enumerated Class by copying the elements/values from the first Enumerated Class, using the “Copy Elements/Values” option on the form. Query the Enumerated Class back using the Query capability of the Enumerated Class form. and modify the Enumerated Class as follows: <ul style="list-style-type: none"><li>• Increase the Valid Length by 1</li></ul>	An ENUMERATED_CLASS_INFO record and the associated ENUMERATED_ELEMENT_INFO records will exist for the Enumerated Class.			

	<ul style="list-style-type: none"><li>• Add an additional element/value pair. Query the Enumerated Class back using the Query capability of the Enumerated Class form.</li></ul>				
3.	<p>Adding/Modifying Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and add a third Enumerated Class by copying the elements/values from the second Enumerated Class, using the “Copy Elements/Values” option on the form. Query the Enumerated Class back using the Query capability of the Enumerated Class form. and modify the Enumerated Class as follows:</p> <ul style="list-style-type: none"><li>• Set the Status to inactive. Query the Enumerated Class back using the Query capability of the Enumerated Class form.</li></ul>	<p>An ENUMERATED_CLASS_INFO record and the associated ENUMERATED_ELEMENT_INFO records will exist for the Enumerated Class.</p>			

4.	Referencing an Enumerated Class - From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and add a DPM FD to the database. From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and attempt to add Hardware information for the DPM FD. associating the third Enumerated Class with the FD.	The operation will fail due to the fact that the third Enumerated Class has an inactive status.	Save the Hardware data for the DPM FD without an associated Enumerated Class.		
5.	Referencing an Enumerated Class - From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and attempt to associate the first Enumerated Class with the DPM FD.	The operation will fail due to the fact that the first Enumerated Class's Valid Length is incompatible with the FD's length.			
6.	Referencing an Enumerated Class - From the Implement option of the CLCS DBSAFE Main Menu, select the Hardware Definition suboption and associate the second Enumerated Class with the Digital Pattern FD. Query the FD back using the Query capability of the Hardware Definition form.	The operation will be successful. A COMMON_HD_INFO record and an associated HD variant record will exist for the DPM FD with the second Enumerated Class association.			
7.	Delete Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and delete the third Enumerated Class and attempt to delete the second Enumerated Class.	The delete to the third Enumerated Class will be successful. The delete of the second Enumerated Class will result in the status for the EC being set to 'I'.			

8.	<p>Modify Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and modify the first Enumerated Class's Valid Length and attempt to modify the second Enumerated Class's Valid Length.</p>	<p>The modify to the first Enumerated Class will be successful. The modify of the second Enumerated Class will fail due to the fact that it is referenced by an FD.</p>			
9.	<p>Modify Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and modify the second Enumerated Class by adding an Element and changing the value of an existing element.</p> <p>A warning message will be presented in an alert box. Select cancel to discard the modify. Query the Enumerated Class back using the Query capability of the Enumerated Class form.</p>	<p>The modify will not have taken place.</p>			
10.	<p>Modify Enumerated Classes - From the DBSAFE Control option of the CLCS DBSAFE Main Menu, select the Enumerated Class suboption and modify the second Enumerated Class by adding an Element and changing the value of an existing element.</p> <p>A warning message will be presented in an alert box. Select ok to accept the modify. Query the EC back using the Query capability of the Enumerated Class form.</p>	<p>The modify will have taken place.</p>			

11.	From the Reports option of the CLCS DBSAFE Main Menu, select the Implement suboption, then select the ONEFD suboption. Request a report of the FDs utilized in the Enumerated Class testing.	The report will contain the specified FDs. The FDs with an associated Enumerated Class will have the enumerated information displayed. The FDs without an associated enumerated class will not have enumerated information displayed.			
12.	From the Implement option of the CLCS DBSAFE Main Menu, select the Lock/Review (DB) suboption. Lock the TPS Run ID used to perform the previous steps. This will automatically generate a From-To-Report.	The From-To Report will be produced for the FDs and will contain the associated enumerated class information.			

Table 3.3

End Time: \_\_\_\_\_

### Signature Page: Test Case 3.3 - Test Enumerated Data Types

_____	_____
<b>Quality Assurance</b>	<b>Date</b>

_____	_____
<b>Test Conductor</b>	<b>Date</b>

**Comments:**

---

---

---

---

---

---

---

---

---

---

### **3.4 TEST CASE 3.4 - TEST COMBINED OI, GPC, PLDOI**

Validate that a combined OI, GPC, PLDOI has been created.

#### **3.4.1 Test Description**

##### **3.4.1.1 Detailed Description**

This test case will demonstrate that a combined OI, GPC, PLDOI has been defined by DBSAFE per the functional requirement stated in section 3.4.1.3. The test case step will demonstrate the existence of the combined gateway as a valid gateway within DBSAFE by using the DBSAFE forms.

##### **3.4.1.2 Resource Requirements**

###### **3.4.1.2.1 Test Personnel**

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

###### **3.4.1.2.2 Hardware**

The following Hardware is required:

- Refer to Section 1.4

###### **3.4.1.2.3 Software**

The following Software is required:

- Refer to Section 1.4

###### **3.4.1.2.4 Data**

The following Data is required:

- Refer to Section 1.4

##### **3.4.1.3 Requirements Summary**

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.17	Validate that a combined OI, GPC, PLDOI has been created. <ul style="list-style-type: none"><li>• Create a OFIA/OFIS gateway supporting data from the GPC Downlist, OI Downlink, and Payload OI Link</li></ul>



### **3.4.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.4.3 Procedure**

Refer to Procedure 3.4 in Table 3.4 for test procedures.

Procedure 3.4 - Test combined OI, GPC, PLDOI					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	From the TCID option of the CLCS DBSAFE Main Menu, select the Gateway suboption and query the defined Gateways using the query capability of the Gateway form.	The OFIA Gateway record will exist.			

Table 3.4

End Time: \_\_\_\_\_

**Signature Page: Test Case 3.4 - Test combined OI, GPC, PLDOI**

---

**Quality Assurance**

---

**Date**

---

**Test Conductor**

---

**Date**

**Comments:**

---

---

---

---

---

---

---

---

---

---

### 3.5 TEST CASE 3.5 - TEST DUAL ADDRESSES

Support for dual FD addresses will be provided.

#### 3.5.1 Test Description

##### 3.5.1.1 Detailed Description

This test case will demonstrate that DBSAFE support for Dual FD Addresses has been provided per the functional requirement stated in section 3.5.1.3. Each step of the test case will demonstrate specific aspects of this support ranging from manipulating Function Designator data in the database using the DBSAFE forms, to running DBSAFE reports.

##### 3.5.1.2 Resource Requirements

###### 3.5.1.2.1 Test Personnel

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

###### 3.5.1.2.2 Hardware

The following Hardware is required:

- Refer to Section 1.4

###### 3.5.1.2.3 Software

The following Software is required:

- Refer to Section 1.4

###### 3.5.1.2.4 Data

The following Data is required:

- Refer to Section 1.4

##### 3.5.1.3 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.19	Support for dual FD addresses will be provided. <ul style="list-style-type: none"> <li>• Allow specification of a second GPC Port /BTU Address (different from the primary address) for LDB/UPLINK type DPSD subtype TWO FDs with a BTU class of MDM or FLX.</li> </ul>

### **3.5.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.5.3 Procedure**

Refer to Procedure 3.5 in Table 3.5 for test procedures.

Procedure 3.5 - Test Dual Addresses					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and an FD with a source of LDB or UPLK, a type of DPSD, a subtype of TWO, and a BTU Class of MDM or FLX to the database. From the Implement option of the CLCS DBSAFE Main Menu, select the Address LDB/Uplink suboption and add Address information for the DPSD FD, including A2 information. Query the FD address information back using the Query capability of the LDB/Uplink Address form.	A MDM_AD_INFO record will exist for the DPSD FD.			
2.	From the Reports option of the CLCS DBSAFE Main Menu, select the Implement suboption, then select the ONEFD suboption. Request a report of the FDs utilized in the Dual Address testing.	The report will contain the specified FD. Address information for the specified FD will be displayed in the report, including A2 data.			
3.	From the Implement option of the CLCS DBSAFE Main Menu, select the Lock/Review (DB) suboption. Lock the TPS Run ID used to perform the previous steps. This will automatically generate a From-To-Report.	The From-To Report will be produced for the FD and will contain Address information, including A2 data.			

Table 3.5

End Time: \_\_\_\_\_

**Signature Page: Test Case 3.5 - Test Dual Addresses**

\_\_\_\_\_  
**Quality Assurance**

\_\_\_\_\_  
**Date**

\_\_\_\_\_  
**Test Conductor**

\_\_\_\_\_  
**Date**

**Comments:**

---

---

---

---

---

---

---

---

---

---

**3.6 TEST CASE 3.6 - TEST CONVERTED TYPE FIELD**

Validate that a Converted Type field has been added.

**3.6.1 Test Description****3.6.1.1 Detailed Description**

This test case will demonstrate that DBSAFE support for a Converted Type field has been provided per the functional requirement stated in section 3.6.1.3. Each step of the test case will demonstrate specific aspects of this support ranging from viewing table structures/data, to manipulating Function Designators in the database using the DBSAFE forms.

**3.6.1.2 Resource Requirements****3.6.1.2.1 Test Personnel**

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

**3.6.1.2.2 Hardware**

The following Hardware is required:

- Refer to Section 1.4

**3.6.1.2.3 Software**

The following Software is required:

- Refer to Section 1.4

**3.6.1.2.4 Data**

The following Data is required:

- Refer to Section 1.4

**3.6.1.3 Requirements Summary**

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.20	Validate that a Converted Type field has been added. <ul style="list-style-type: none"> <li>• Provide a lookup table for association of FD type to Converted Type.</li> <li>• Add the Converted Type Field to the FD Compiler record.</li> </ul>



### **3.6.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.6.3 Procedure**

Refer to Procedure 3.6 in Table 3.6 for test procedures.

Procedure 3.6 - Test converted type field					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Via SQL*Plus, a describe of the DBSAFE.FD_LENGTH_INFO and DBSAFE.COMMON_CD_INFO tables will be performed.	A CONVERTED_TYPE column will be present in both tables.			
2.	Via SQL*Plus, a select of distinct TYPE and CONVERTED_TYPE from the DBSAFE.FD_LENGTH_INFO table will be performed.	For every distinct TYPE there is an associated CONVERTED_TYPE in the table.			
3.	From the Implement option of the CLCS DBSAFE Main Menu, select the Compiler Definition suboption and add a DM FD with to the database. Query the FD back using the Query capability of the Compiler Definition form.	A COMMON_CD_INFO record will exist for the DM FD with the CONVERTED_TYPE field valued.			

Table 3.6

End Time: \_\_\_\_\_

**Signature Page: Test Case 3.6 - Test converted type field**

<b>Quality Assurance</b>	<b>Date</b>

<b>Test Conductor</b>	<b>Date</b>

**Comments:**

### 3.7 TEST CASE 3.7 - TEST CALIBRATION FDS

Validate that support for Calibration FDs for LDB Analogs has been provided.

#### 3.7.1 Test Description

##### 3.7.1.1 Detailed Description

This test case will demonstrate that DBSAFE support for Calibration for LDB Analog FDs has been provided per the functional requirement stated in section 3.7.1.3. Each step of the test case will demonstrate specific aspects of this support ranging from viewing table structures/data, to manipulating Function Designators in the database using the DBSAFE forms.

##### 3.7.1.1.1 Test Personnel

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

##### 3.7.1.1.2 Hardware

The following Hardware is required:

- Refer to Section 1.4

##### 3.7.1.1.3 Software

The following Software is required:

- Refer to Section 1.4

##### 3.7.1.1.4 Data

The following Data is required:

- Refer to Section 1.4

#### 3.7.1.2 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.21	<p>Validate that support for Calibration FDs for LDB Analogs has been provided.</p> <ul style="list-style-type: none"> <li>• Remove M/B Scaling from the calibration FD.</li> <li>• Remove support for Spare Curves w/o Data (type LSS).</li> <li>• Remove support for Single Segment curves (type LS1) and Single Segment Spare Curves (type LSS1),</li> <li>• Remove support for valid_converted length associated with Cal FDs.</li> <li>• Support normal calibration on LDB.</li> </ul>

### **3.7.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.7.3 Procedure**

Refer to Procedure 3.7 in Table 3.7 for test procedures.

Procedure 3.7 - Test Calibration FDs					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Via SQL*Plus, a describe of the DBSAFE.CALIBRATION_CD_INFO table will be performed.	The M_SCALING, B_SCALING, and VALID_CONVERTED_LENGTH columns will not exist.			
2.	Via SQL*Plus, a select of TYPE (order by type) from the DBSAFE.TYPE_INFO table will be performed.	Types LS1, LSS, and LSS1 will not be contained in the table.	Therefore, they are no longer legal.		
3.	From the Implement option of the CLCS DBSAFE Main Menu, select the Cal FD Definition suboption and add a LS4 Calibration FD with to the database. Query the FD back using the Query capability of the Cal FD Definition form.	A CALIBRATION_CD_INFO record will exist for the LS4 FD.	During the test it will be observed that the removed types - LSS, LS1, and LSS1 are no longer valid using the Calibration Definition form.		
4.	From the Implement option of the CLCS DBSAFE Main Menu, select the Cal FD Definition suboption and add a LSS4 Calibration FD with to the database. Query the FD back using the Query capability of the Cal FD Definition form.	A CALIBRATION_CD_INFO record will exist for the LSS4 FD.			

Table 3.7

End Time: \_\_\_\_\_

**Signature Page: Test Case 3.7 - Test Calibration FDs**

<hr/>	<hr/>
<b>Quality Assurance</b>	<b>Date</b>

<hr/>	<hr/>
<b>Test Conductor</b>	<b>Date</b>

**Comments:**

---

---

---

---

---

---

---

---

---

---

### 3.8 TEST CASE 3.8 - TEST SYSTEM FDS

Validate that support to include all System FDs in the FD Directory and OLDB has been provided.

#### 3.8.1 Test Description

##### 3.8.1.1 Detailed Description

This test case will demonstrate that DBSAFE support for including all System FDs in the FD Directory and OLDB has been provided per the functional requirement stated in section 3.8.1.3. The test case step will demonstrate the existence of the table view to support CLCS Test Build.

##### 3.8.1.2 Resource Requirements

###### 3.8.1.2.1 Test Personnel

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

###### 3.8.1.2.2 Hardware

The following Hardware is required:

- Refer to Section 1.4

###### 3.8.1.2.3 Software

The following Software is required:

- Refer to Section 1.4

###### 3.8.1.2.4 Data

The following Data is required:

- Refer to Section 1.4

##### 3.8.1.3 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.22	Validate that support to include all System FDs in the FD Directory and OLDB has been provided. <ul style="list-style-type: none"><li>• Add a view to support CLCS Test Build</li></ul>



### **3.8.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.8.3 Procedure**

Refer to Procedure 3.8 in Table 3.8 for test procedures.

Procedure 3.8 - Test System FDs					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Via SQL*Plus, a describe of the TBL_SYSTEM_STATUS_HD_INFO view will be performed.	The TBL_SYSTEM_STATUS_HD_INFO view will exist.			

Table 3.8

**End Time:** \_\_\_\_\_

**Signature Page: Test Case 3.8 - Test System FDs**

\_\_\_\_\_  
**Quality Assurance**

\_\_\_\_\_  
**Date**

\_\_\_\_\_  
**Test Conductor**

\_\_\_\_\_  
**Date**

**Comments:**

---

---

---

---

---

---

---

---

---

---

### 3.9 TEST CASE 3.9 - TEST IVHM FDS

Validate that support for IVHM FDs has been provided.

#### 3.9.1 Test Description

##### 3.9.1.1 Detailed Description

This test case will demonstrate that DBSAFE support for IVHM FDs has been provided per the functional requirement stated in section 3.9.1.3. Each step of the test case will demonstrate specific aspects of this support ranging from viewing table structures/data, to verifying the existence of data in the Database using the DBSAFE forms.

##### 3.9.1.2 Resource Requirements

###### 3.9.1.2.1 Test Personnel

Personnel required include at a minimum a Test Conductor, a QA witness and I&T. Skills required by the test conductor (or designee) include an understanding of the CLCS DBSAFE execution environment and general Oracle knowledge.

###### 3.9.1.2.2 Hardware

The following Hardware is required:

- Refer to Section 1.4

###### 3.9.1.2.3 Software

The following Software is required:

- Refer to Section 1.4

###### 3.9.1.2.4 Data

The following Data is required:

- Refer to Section 1.4

##### 3.9.1.3 Requirements Summary

This test case demonstrates that the following functional requirements are met:

Requirement Number	Description
1.2.2.23	Validate that support for IVHM FDs has been provided. <ul style="list-style-type: none"> <li>• Add a new source (SRC) - CSGW.</li> <li>• Add a new link/gateway - CS/CSGW</li> </ul>

### **3.9.2 Pass/Fail Criteria**

Successful completion of the test procedures without any problems that would result in the generation of critical problem reports and without an excessive number of major problem reports will be sufficient for this test to be considered passed.

### **3.9.3 Procedure**

Refer to Procedure 3.9 in Table 3.9 for test procedures.

Procedure 3.9 - Test IVHM FDs					
		Date:	Location:	Start Time:	
Test Setup/Initial Conditions - Refer to Appendix D					
Step	Description	Expected Results	Comments	TC	QA
1.	Via SQL*Plus, a select of all sources from the DBSAFE.SOURCE_INFO table will be performed.	The Consolidated System Gateway source, CSGW, will be present.			
2.	From the TCID option of the CLCS DBSAFE Main Menu, select the Gateway suboption and query the defined Gateways using the query capability of the Gateway form.	The CSGW Gateway record will exist.			

Table 3.9

**End Time:** \_\_\_\_\_

**Signature Page: Test Case 3.9 - Test IVHM FDs**

---

**Quality Assurance**

---

**Date**

---

**Test Conductor**

---

**Date**

**Comments:**

---

---

---

---

---

---

---

---

---

---

## Appendix A Acronyms and Definitions

Reference Document	Document Number	Rev.	Date
CLCS Acronyms	84K00240	Basic	11/13/1997
CLCS Glossary	84K00250	Basic	11/13/1997



## Appendix B Requirements Traceability and Test Methods Matrix

The following table is intended to show which CLCS Functional Requirement is demonstrated in each CLCS DBSAFE CSCI Integration Test (CIT) and what test method was used in that test case. This table will be updated and baselined with each CIT starting with the Redstone Delivery.

Functional Requirement	Traced SLS Requirement	CI Test	Test Case	Test Method			
				Inspection	Analysis	Demo	Test
1.2.2.14	2.4.2.1, 2.4.2.2	Thor CIT	3.1			✓	✓
1.2.2.15	2.4.2.1, 2.4.2.2	Thor CIT	3.2			✓	✓
1.2.2.16	2.4.2.1, 2.4.2.2	Thor CIT	3.3				✓
1.2.2.17	2.4.2.1, 2.4.2.2	Thor CIT	3.4				✓
1.2.2.19	2.4.2.1, 2.4.2.2	Thor CIT	3.5				✓
1.2.2.20	2.4.2.1, 2.4.2.2	Thor CIT	3.6			✓	✓
1.2.2.21	2.4.2.1, 2.4.2.2	Thor CIT	3.7			✓	✓
1.2.2.22	2.4.2.1, 2.4.2.2	Thor CIT	3.8			✓	
1.2.2.23	2.4.2.1, 2.4.2.2	Thor CIT	3.9			✓	✓

## Appendix C Resource Requirements

Refer to Section 1.4

## Appendix D Standard Test Operating Procedures

To gain access to DBSAFE, one must first have a valid userid on the LSDN. Contact the LSDN Help Desk to establish a new account and to obtain any necessary training on how to use a workstation. The Help Desk provides information to help resolve connectivity issues for persons without local access to an LSDN workstation.

Next, obtain an RSI password through the Access Control Data Base Administrator (ADBA) in the USA Quality group. The ADBA must update the RSI authorized access list, establish the Oracle account, and grant authorization to use the DBSAFE\_USER database role. DBSAFE\_USER is a password protected role that provides insert, update, and delete privileges on data and execute privileges on database procedures owned by DBSAFE. When an authorized person initiates DBSAFE, the main menu looks up the password and sets their role to DBSAFE\_USER. This provides the rights needed to modify the DBSAFE database while running DBSAFE applications. This role assignment is terminated when the DBSAFE session ends.

The final steps to obtain access to DBSAFE are managed by the DBSAFE Administrator. This is the person(s) authorized to run the Maintain User Data form to define authorized DBSAFE operators, their rights to execute controlled DBSAFE functions, and their rights to change or delete specific sets of data within the DBSAFE Database. This data is initially checked by the DBSAFE main menu to ensure only authorized people running authorized functions are allowed to modify data. DBSAFE applications access this data as needed to continually enforce privilege constraints defined by the DBSAFE Administrator.

With the following prerequisites met, CLCS DBSAFE can be initiated by typing `rXsi -c f40runmx dbsafec` (with X being d for Development, v for Validation or o for Operations) from a UNIX prompt on the LSDN. The CLCS DBSAFE Main Menu will be displayed, from which the numerous CLCS DBSAFE functions are available.